DEPARTMENT OF DEFENCE

DSTO

Information Technology Division

RESEARCH NOTE
ERL-0593-RN

REAL TIME CHARACTER RECOGNITION FOR F-111C VIDEO TAPE RECORDS

by

Stephen Bourn

APPROVED FOR PUBLIC RELEASE

ELECTRONICS RESEARCH LABORATORY

AR-006-924

DSTO
AUSTRALIA

ELECTRONICS RESEARCH LABORATORY

Information Technology Division

RESEARCH NOTE
ERL-0593-RN

REAL TIME CHARACTER RECOGNITION FOR F-111C VIDEO TAPE RECORDS

by

Stephen Bourn

SUMMARY

For evaluation purposes it is often necessary to reconstruct the tracks of F-111C aircraft participating in Australian Defence Force Exercises. The necessary information can be obtained from a video tape recording of one of the aircraft's displays. An image processing system has been developed to automatically extract this data.

NOV 93

APPROVED FOR PUBLIC RELEASE

POSTAL ADDRESS: Director, Electronics Research Laboratory, PO Box 1500, Salisbury, South Australia, 5108.

ERL-0593-RN

ABBREVIATIONS

ADC	Analogue to Digital Converter
DAC	Digital to Analogue Converter
LUT	Look Up Table

THIS IS A BLANK PAGE

CONTENTS

FIGURES

THIS IS A BLANK PAGE

1 INTRODUCTION

Staff in Exercise Analysis Group, Information Technology Division, regularly monitor military exercises conducted by the Australian Defence Force. When aircraft are involved it is often necessary to obtain detailed and precise information on the aircraft's track. In the case of the F-111C, the necessary data can be extracted from a video tape recording of one of the aircraft's graphical displays. The display can be switched to show a radar image or an infra-red image. When the radar image is selected, two columns of data also appear on either side, as shown in Figure 1. The date and time appear at the bottom left of both images. The latitude, longitude and altitude from the data columns, together with the time, enable the track to be reconstructed.

This paper describes a system that has been developed to automatically extract the track information from the video tape. The system has been successfully used to reconstruct F-111C tracks for a number of Air Defence exercises. An overview of the system has been described in the proceedings of a conference [Ref. 1].

2 EARLY DEVELOPMENT

Reference 2 describes the findings of an initial feasibility study. This included a review of the literature on character recognition. Current research interest is in extracting transformation invariant features to enable recognition of handwritten and multi font characters, which vary in size and rotation, and suffer from other distortions. Powerful, general recognition algorithms suffer the drawback of requiring lengthy, numerically intensive calculations. References 3 and 4 describe the first system which was developed by staff in Exercise Analysis Group. It used an algorithm specifically designed for the characters displayed on the F-111C video tapes. The algorithm extracted a small but sufficient set of basic features which enabled the characters to be distinguished. This resulted in short computation times to recognise each character, and so enabled frequent sampling and processing of video frames as tapes were replayed at normal speed.

Unfortunately this first system was not flexible enough to cope with the variation which occurred in the video images. The error rate was too high and the system was never successfully used. This paper describes a second system which has been developed. The concepts used in the first system were refined, a different and more robust set of character features were chosen, and other variants in the image were dynamically catered for. The code was completely rewritten. As stated in section 1, this system has now been successfully used a number of times.

3 THE RECOGNITION SYSTEM

Figure 2 shows a block diagram of the system. A PCVISIONplus Frame Grabber board has been installed in a PC compatible computer. The recording is played at normal speed and the video signal made available to the Frame Grabber. Under control of the main loop of the program a single frame is acquired, digitised and stored in frame memory. The stored image is then inspected and the data recognised and saved. Figure 3 shows a flow chart of the program. Processing a single frame takes a few seconds on a computer with XT performance, after which the next available video frame is acquired.

4 OPERATION OF THE PCVISIONplus FRAME GRABBER

Reference 5 gives a full description of the Frame Grabber operation. A brief overview is given here.

The video signal for a single frame consists of 480 sections of continuous analogue information, one section for each line of the display. Each section is preceded by a horizontal blank section which aids synchronisation. A typical portion of the video signal is shown in Figure 4. Lines can be thought of as separate but the information within each line is continuous. The programmable gain was set to give maximum boost to the incoming signal, and the offset was programmed to add the maximum voltage to the signal which still leaves the background black.

The analogue to digital converter (ADC) samples each line 512 times, and produces 8 bit digital values, which will be called digitised pixels.

A look up table (LUT) is used to transform digitised pixel values. It is implemented as a list of 256 bytes and can be programmed to represent any transformation mapping 8 bit values to 8 bit values. The input value is used as an address to look up the output value in the list. The input LUT was set to be the ramp function or identity transformation.

Output from the input LUT is stored in frame memory. The digitised image is held in frame memory as 480x512 digitised pixels. Frame memory architecture allows simultaneous acquisition and display of images. For display the digitised pixel information is transformed by the output LUTs and converted by three digital to analogue converters (DACs). A single Frame Grabber board accepts only one input signal and a bank of three boards would be needed for true colour image processing.

Communication with the host is via the control registers which occupy 16 bytes in the host I/O space and frame memory occupies 64 KBytes in the host memory address space. A pixel buffer reduces contention between the boards continuous display requirements and host access. The host can set the offset and gain, program the LUTs, select the image acquisition mode to be continuous or otherwise, and read or write to any digitised pixel in frame memory.

5 CHARACTERISTICS OF THE DIGITISED IMAGE

Although nothing is known of the systems which generate the displayed alphanumeric data it is useful to develop a working model which explains the observed characteristics of the digitised images.

The characters seem to have been logically designed on a 7x5 array of logical pixels. Some examples are shown in Figure 5. Consider the top row of the zero. As the logical row is scanned from left to right the continuous video signal must be generated. The logic is low for one logical pixel, goes high for 3 logical pixels and then returns to the low state. With instantaneous response one would expect the corresponding portion of the analogue video signal to be 0 volts for a period of time equivalent to one logical pixel, then jump to some positive level for three logical pixels, and then jump back to 0 volts. However response is not instantaneous. Figure 6 gives two examples showing that if the logic goes high at time zero, the maximum voltage, measured in terms of the digitised intensity, is not achieved until the time equivalent of two logical pixels has elapsed. Figure 7 gives two examples showing the response when the logic goes low.

This means that the maximum voltage is not achieved unless at least two adjacent logical pixels in a row have logic high. Figure 8 shows examples of logical pixel patterns that remain if isolated logical pixels in rows are ignored.

Figure 9 shows an example of the digitised pixels of a zero. The 8 bit intensity is written in each digitised pixel square. Recall that the term digitised pixel is used to indicate one of the 480x512 digital values produced for each frame and is not to be confused with the term logical pixel. Notice that precisely two rows of digitised pixels correspond to each logical row. Across a row, logical pixel design is converted to a continuous analogue signal which is then sampled to produce the digitised pixels. There are about 1.75 digitised pixels per logical pixel, but the respective lengths are almost certainly incommensurable.

6 RECOGNITION OF DIGITISED IMAGE

A threshold can be found such that the intensity in rows with no two adjacent logical pixels remains below the threshold, while in rows with two or more adjacent logical pixels the intensity climbs above the threshold at some stage. For the example in Figure 9 a threshold of 140 could be used. The intensity is written in black characters in the digitised pixel squares which have intensity equal to or above the threshold. Compare the enclosed regions with the pattern shown in Figure 8. Observation of which rows achieve the threshold is sufficient to distinguish the zero digit from any other digit.

The function *digit_or_blank* which recognises the digits 0,1,2, ...,9 and the space character, represented by b for blank, can be represented by a decision tree as shown in Figure 10. At each internal node the possible values are shown in bold and the logical test to be performed is shown in normal type. The tree is descended until a leaf is reached, by which time the character has been determined.

The function *blank* which returns true or false is used to determine whether a particular logical row achieves the threshold. A maximum of four applications of *blank* is sufficient to recognise the digits 0, 1, 4, 5 or 7 or the space character.

The digits 2, 3, 6, 8 and 9 all achieve the threshold in logical rows 1, 4 and 7. To distinguish these characters three new functions are needed. The function *length* returns the number of digitised pixels in a logical row which achieve the threshold. The function *first* returns the number of digitised pixels to the first digitised pixel which achieves the threshold and the function *last* returns the number of digitised pixels to the last digitised pixel which achieves the threshold. Because the relationship between logical pixels and digitised pixels is not entirely predictable it is necessary to look for gross characteristics to distinguish characters. For example the logical length of row 1 of the digit three is 3 logical pixels greater than row 4. Two calls to *length* are sufficient to recognise the digit three. The first logical pixel in row 1 of the digit six occurs two logical pixels later than the first logical pixel in row 4. By looking for characteristics in this manner the digits 3, 2, 6 and 9 are recognised, leaving the digit eight by default.

Logical row four of the digit zero occasionally achieves the threshold and otherwise redundant tests in the recognition logic allow for this.

Where alphabetical characters occur there are only ever a few possibilities to choose between. For example in the latitude one call to *blank* can distinguish between the characters N and S.

The algorithm described in this section uses a minimal but sufficient and robust set of features to recognise the characters, following a minimum necessary number of logical tests. This means that the implementation is very fast, and a high throughput can be achieved. A high throughput could not have been achieved using a more general character recognition algorithm.

7 OTHER PROGRAM FEATURES

The signal coming in to the Frame Grabber board can be modified by the programmable offset and gain prior to being digitised by the ADC. Gain refers to amplification of the incoming signal, while offset refers to a constant voltage that is added to the entire signal. Recall that the gain is set to give the maximum boost to the incoming signal, while the offset is required to add the maximum voltage to the signal which still leaves the background black. The program must initially determine the required offset, as this varies with different video players, or when an optional monitor is connected. This is done automatically by systematic trial and error. Offset is set to the maximum value, a frame is acquired, and the background is tested for zero intensity. This process is repeated, decrementing the offset each time, until the background does have zero intensity.

When a frame is acquired it is necessary to first check whether it is a radar image. In the data columns which appear on the radar image certain characters are always present. The digitised pixels in these characters would have high intensity while the immediately adjacent digitised

pixels would have zero intensity. When the infra-red image is selected the image fills the whole screen, including the positions otherwise occupied by the data columns. High contrast edges occurring in certain positions within the regions occupied by the data columns are therefore always present in the radar image and extremely unlikely in the infra-red image. In the infra-red image the date and time are still present. When the tape ends the image is blank, in the sense that there is uniform, though not necessarily zero, intensity across the whole image. High contrast in the region occupied by the date and time is therefore common in the infra-red image, but not present when the tape ends. These distinguishing features allow simple but reliable heuristic tests to determine whether the current frame is a radar image. Infra-red images are discarded, and if the end of the tape is detected, the program stops.

The vertical position of data on the image is constant, and the relative position of data items within a line is constant, but the absolute position of the data within lines varies down the screen and from frame to frame. The vertical position of data is determined by its physical line on the display device. In the video signal, lines are very definitely separated by the horizontal blank section. The horizontal position of data is determined by the lag of that part of the signal behind the horizontal synchronisation pulse. The observed variation in the horizontal position of parts of the image is presumably due to variations in the lag between the horizontal synchronisation pulse causing the sampling of the signal to begin for that line, and the data in the signal. For each radar image acquired the program scans horizontally to find the edges of the 16 rows of displayed alphanumeric data.

The maximum intensity varies from frame to frame and for each frame it is necessary to determine the two thresholds to be used for the column data and the generally lower intensity date and time data. A permanently displayed zero digit is inspected to set the column data threshold. A more complex algorithm uses an as yet unrecognised digit in the time group to set the date and time threshold. All digits have two or more adjacent logical pixels in either the first or second logical row, but not in the sixth row. The date and time threshold is set to be a value which is exceeded in either the first or second logical row, but which itself exceeds the intensity of any digitised pixel in the sixth logical row.

The output LUT was programmed as shown in Figure 11. This causes those parts of the image which achieve the thresholds to jump in intensity on the Frame Grabber output monitor. In this way, with very little processing time penalty, the recognition process can be monitored by an observer, which is useful for program debugging, validation, and maintenance.

8 THE PROGRAMMING LANGUAGE AND STYLE

The program was written in C so that use could be made of a library of subroutines which simplify control of the Frame Grabber board. There are over 2000 lines of source code.

The rewritten program is modular and block structured. The use of global variables has been avoided. A functional rather than procedural style has been used, and much of the code is reminiscent of Lisp. This has facilitated development and modification, and the testing of individual modules.

For example Figure 12 shows two optional fragments of code which both recognise the third character in line five of column two and store its value in an array. The functions *side_col_char* and *top_col_char* return the horizontal and vertical location respectively of the left most, topmost digitised pixel of the character. The functional style option was preferred.

9 POSSIBLE SPEED ENHANCEMENT

The recognition algorithm compares digitised pixel intensity to a threshold. A single bit could store the result. The 8x8 bit pixel buffer transfers a byte to the host in a single memory cycle, and the byte represents the intensity of one digitised pixel. An alternative mode allows the transferred byte to be composed of a certain order bit from each of 8 digitised pixels.

The input LUT could be programmed to set the high order bit according to whether intensity was above the threshold. 8 high order bits could be transferred to the host in a single memory cycle and bitwise operations could be performed to recognise the image.

This approach should give a great performance increase at the cost of much programming effort. The simpler approach gives satisfactory performance for the current application.

10 CONCLUSIONS

Careful analysis of the image has enabled the development of an unsophisticated algorithm to recognise alphanumeric characters recorded on a video tape. The number of tests required has been kept to a minimum, giving a short cycle time and allowing a new frame to be processed every few seconds, as the tape is replayed at normal speed.

The algorithm's application is limited to the image for which it was specifically designed, but the general approach and development process may be of wider interest.

11 ACKNOWLEDGMENT

The programming support provided by Peter Duldig is gratefully acknowledged.

REFERENCES

1. Stephen Bourn — "Real Time Character Recognition for F-111C Video Tape Records", IREE Image Processing Conference Proceedings, Canberra, December 1989, pp 87-90.

2. J. Leske and M. Scholz — "Video Recognition System", Group Working Paper, Planning and Data Analysis Group, Trials Branch, AEL, DSTO, 1985.

3. A. Retallick — "Video Data Transcription System User's Guide", Group Working Paper, Planning and Data Analysis Group, Trials Branch, AEL, DSTO, 1987.

4. A. Retallick — "Video Data Transcription System", Group Working Paper, Planning and Data Analysis Group, Trials Branch, AEL, DSTO, 1987.

5. PCVISIONplus Frame Grabber User's Manual, Imaging Technology Incorporated, 1986.

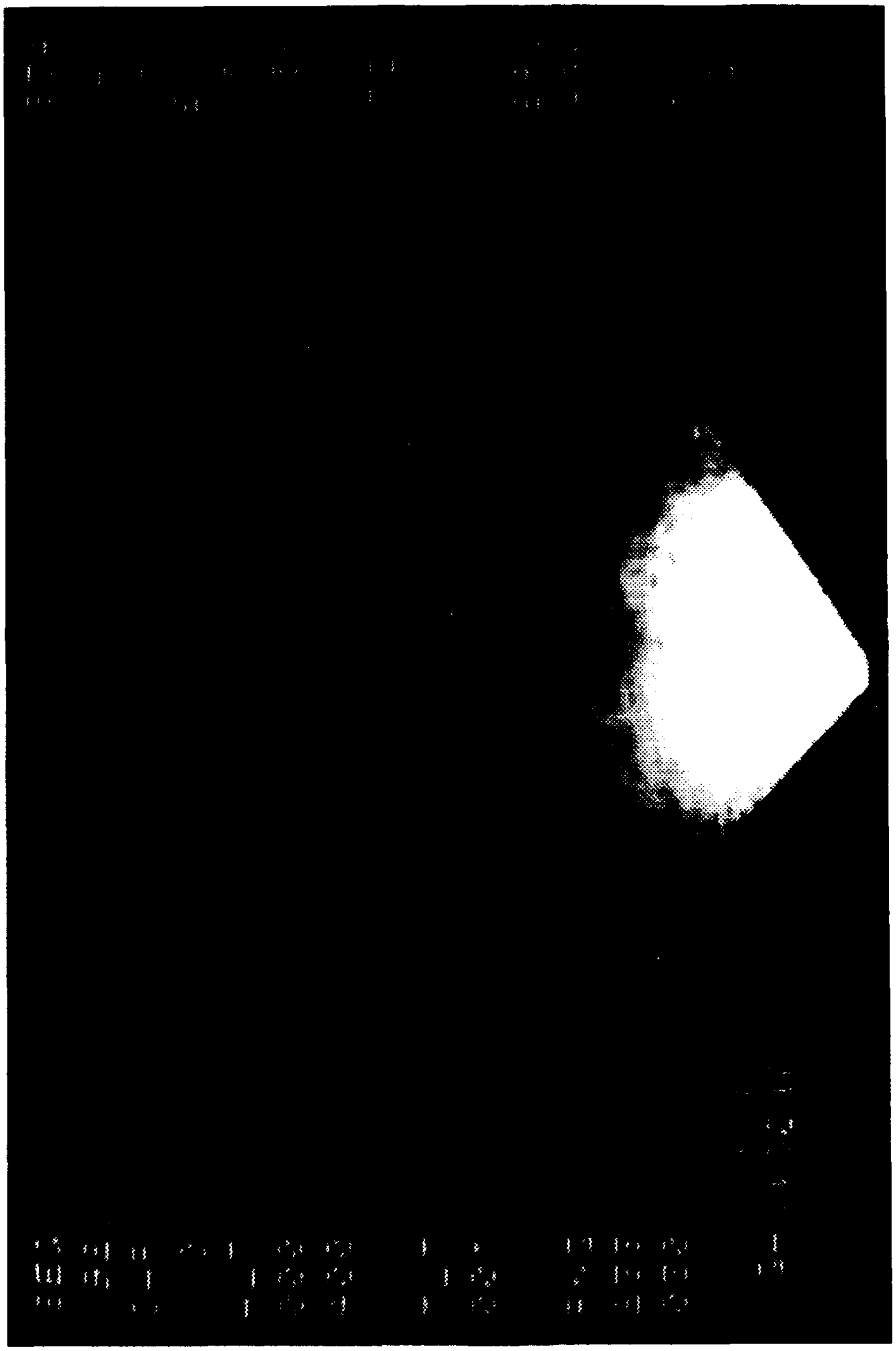

Figure 1 **Original video image**

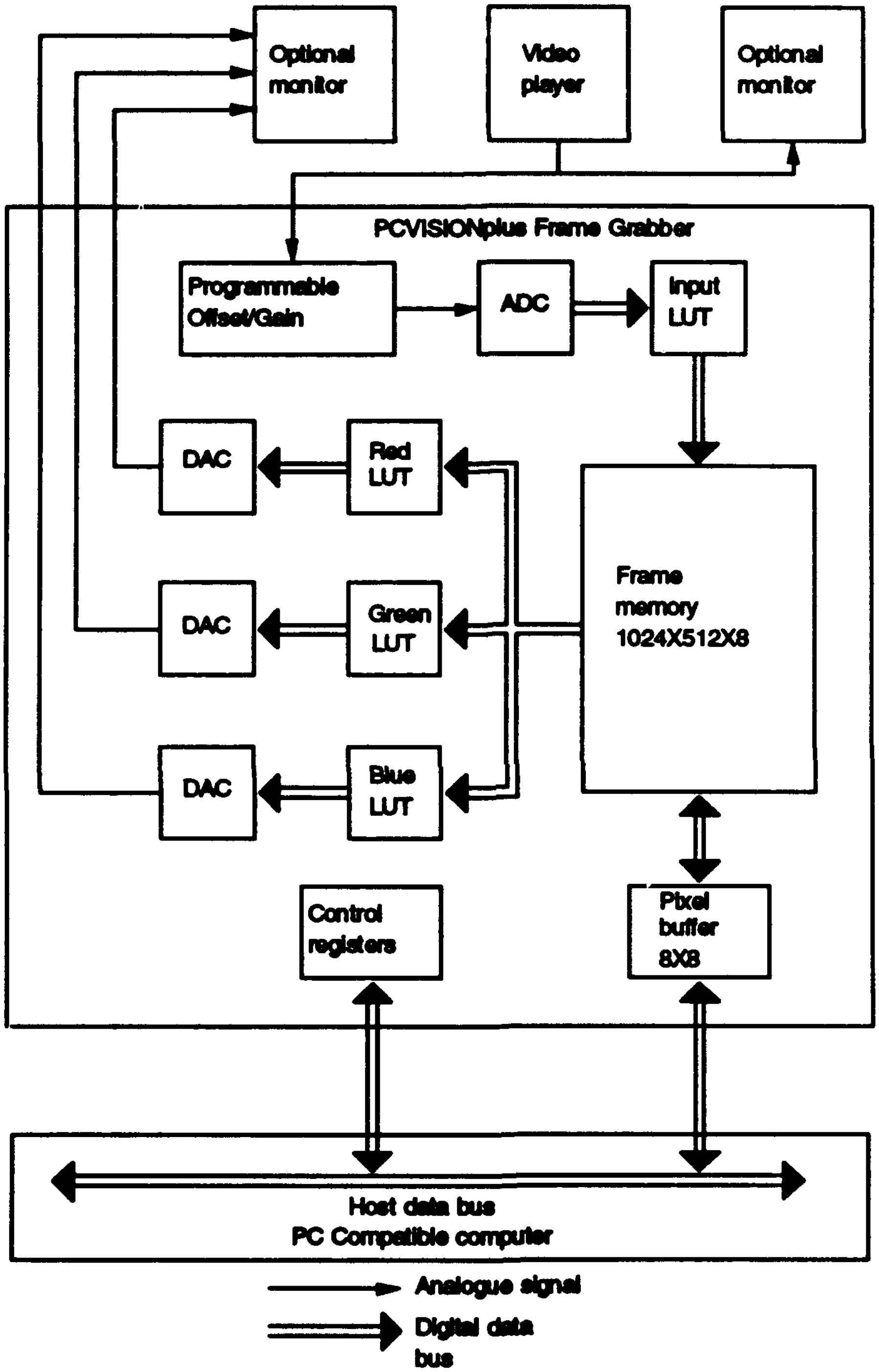

Figure 2 **Block diagram of recognition system**

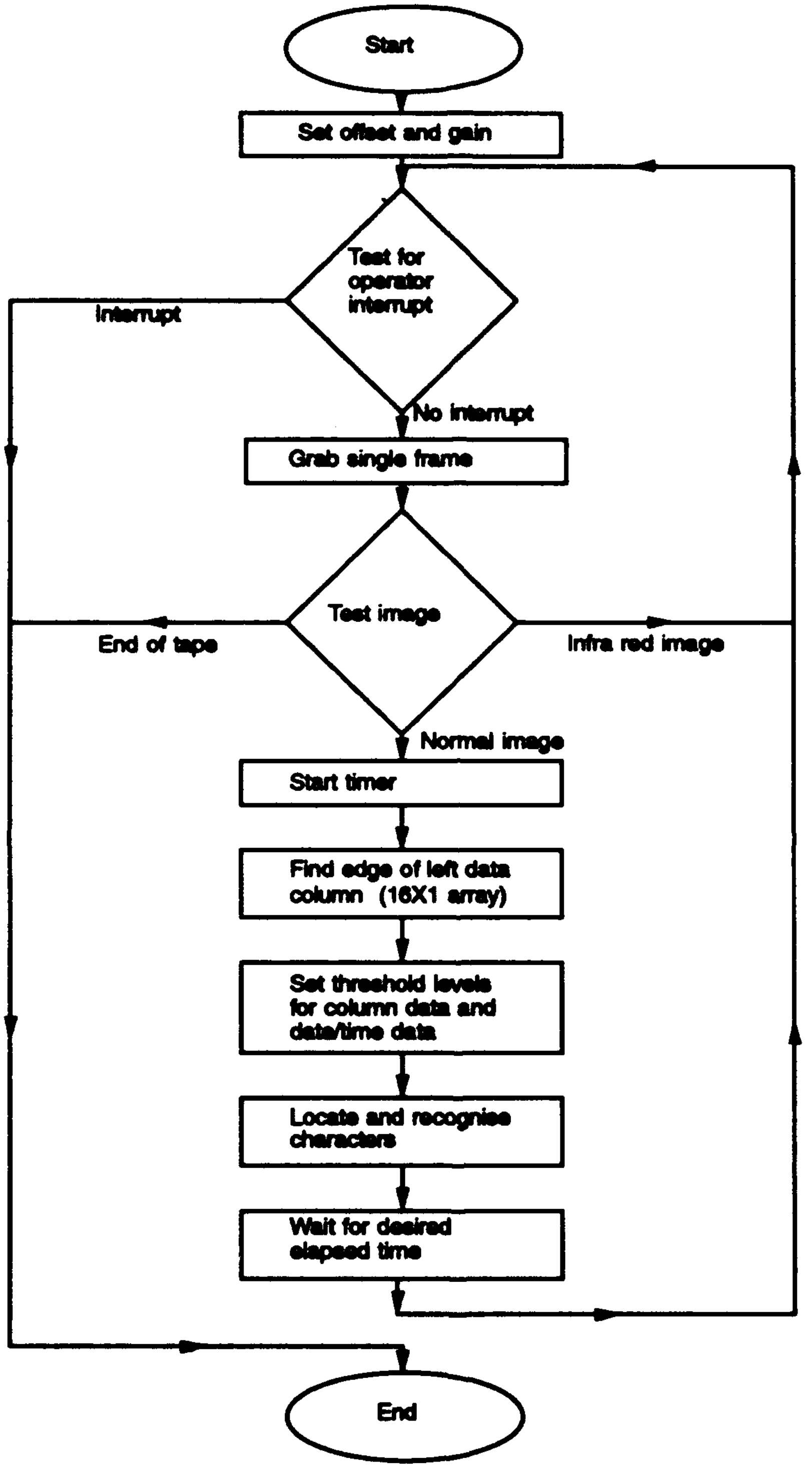

Figure 3 **Flow chart of recognition program**

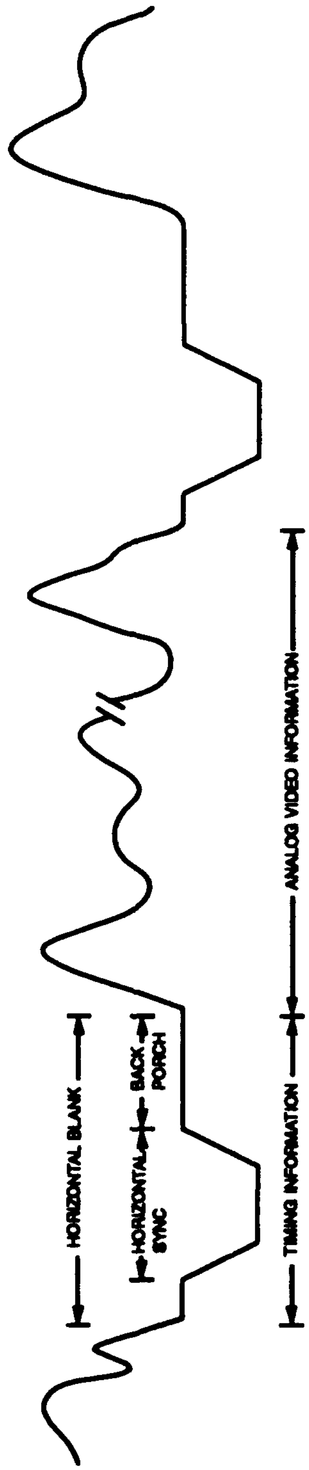

Figure 4 Typical portion of video signal

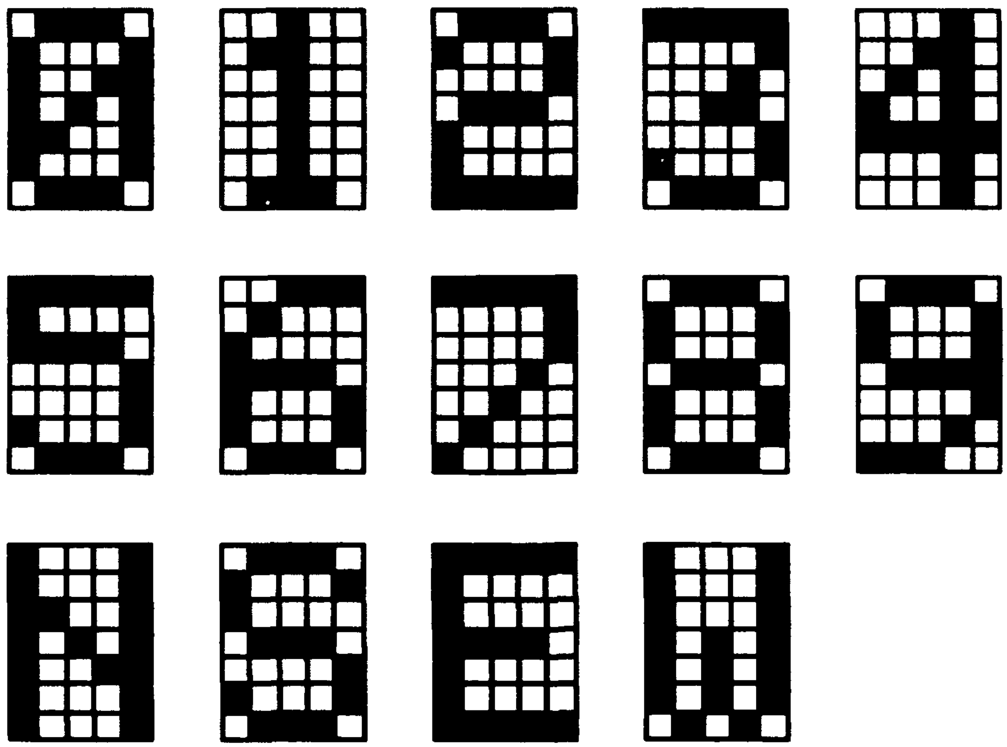

Figure 5 **Examples of logical pixel design of characters**

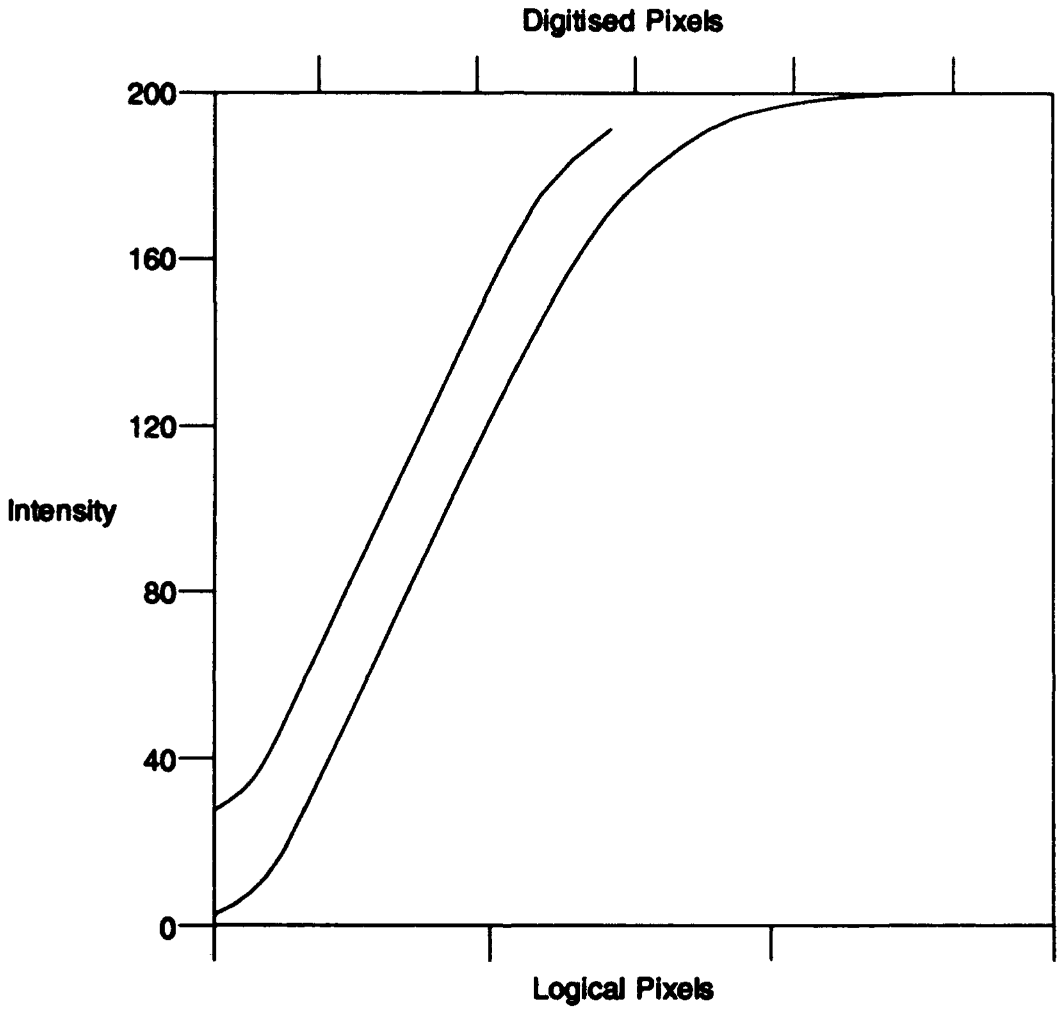

Figure 6 **Two examples of the response of the displayed intensity when the logic goes high**

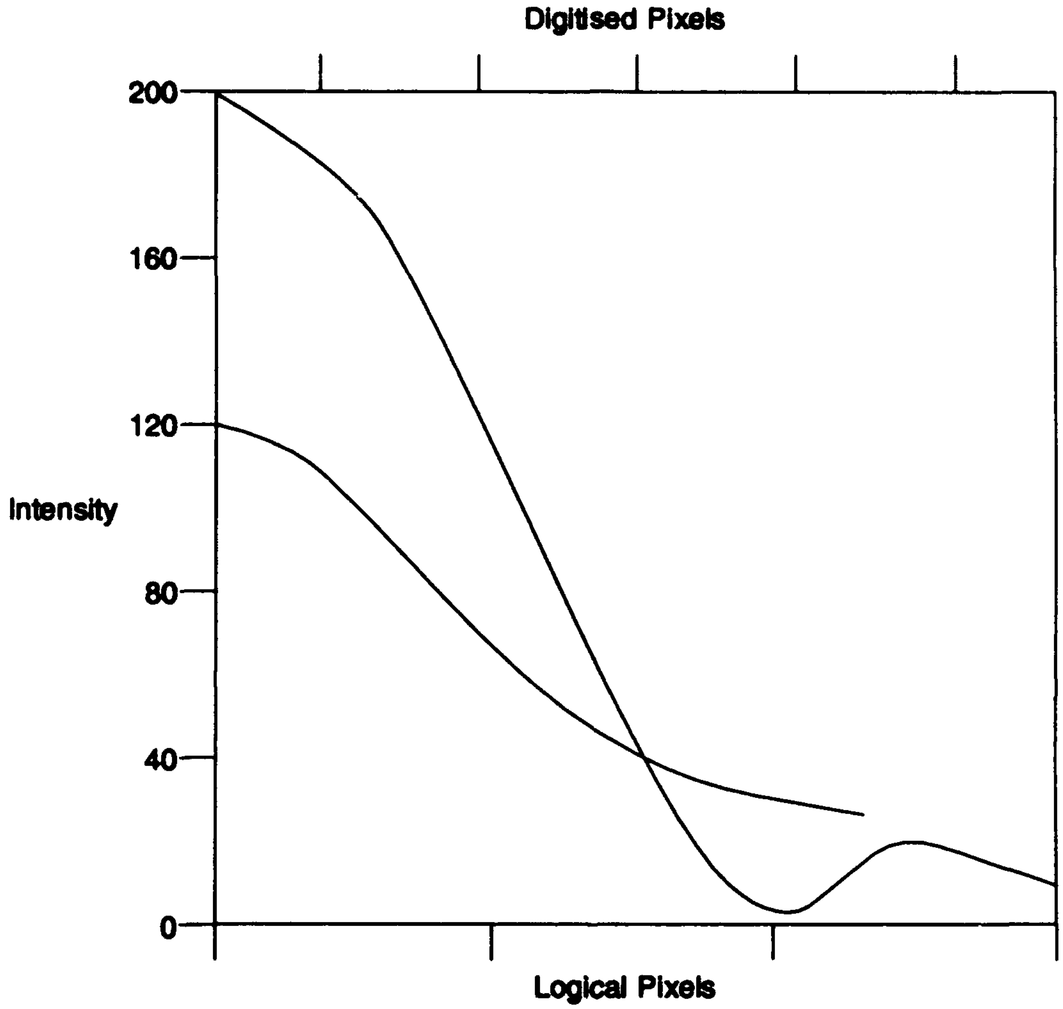

Figure 7 Two examples of the response of the displayed intensity when the logic goes low

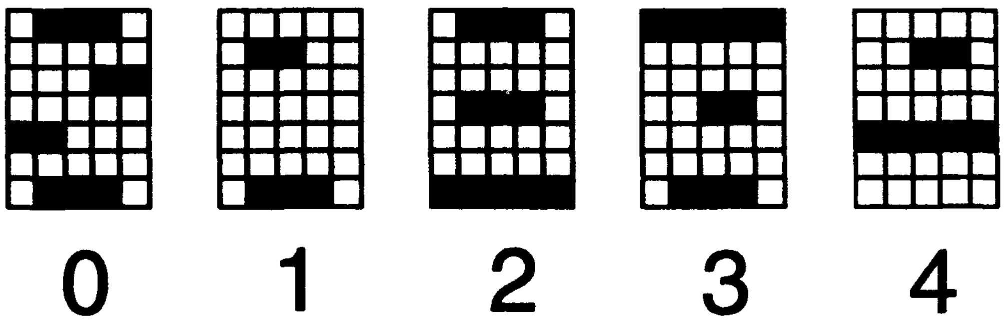

Figure 8 Examples of logical pixels with intensity expected to be above the threshold

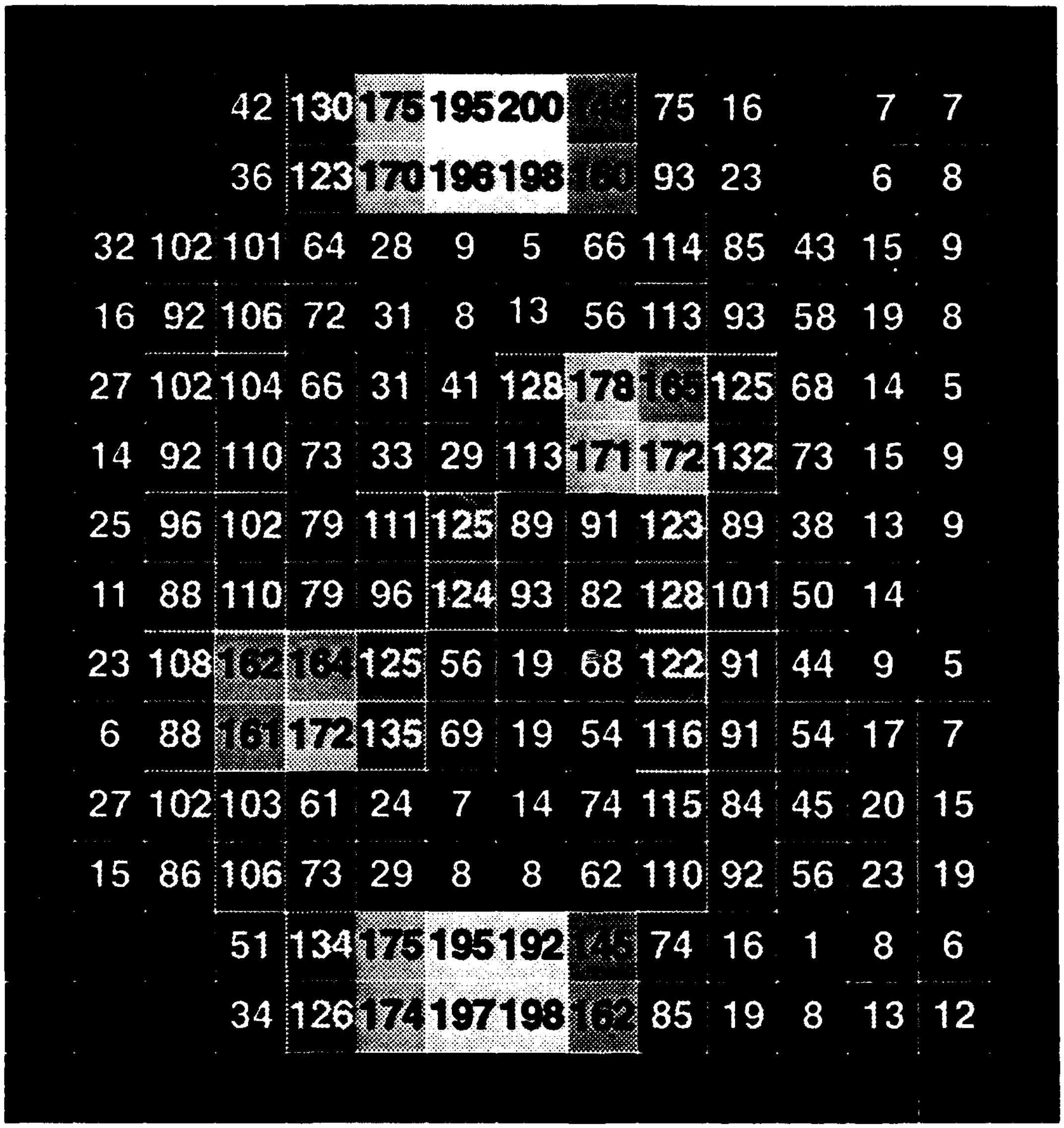

Figure 9 **Digitised zero showing intensity of digitised pixels**

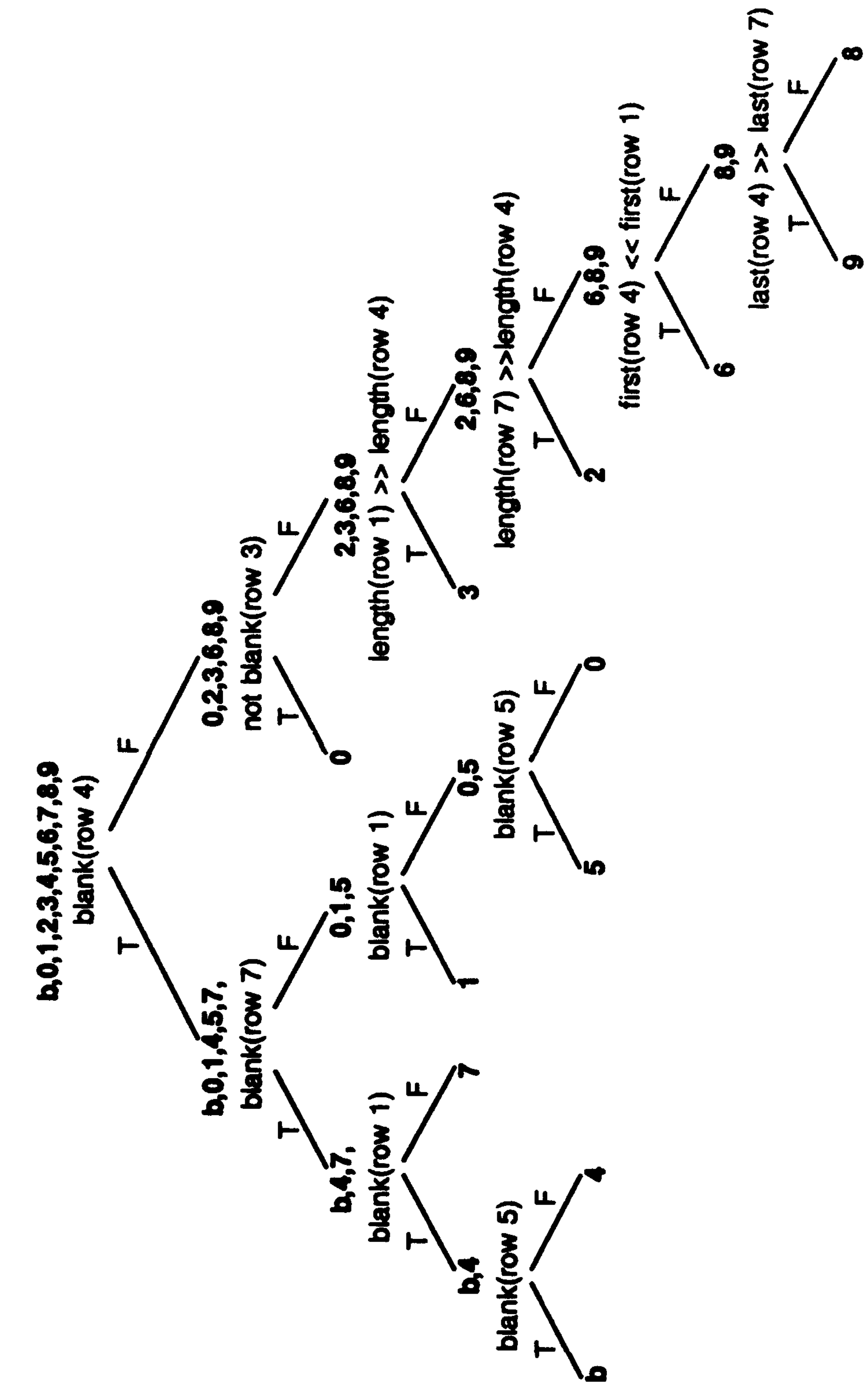

Figure 10 **Decision tree for the function *digit_or_blank***

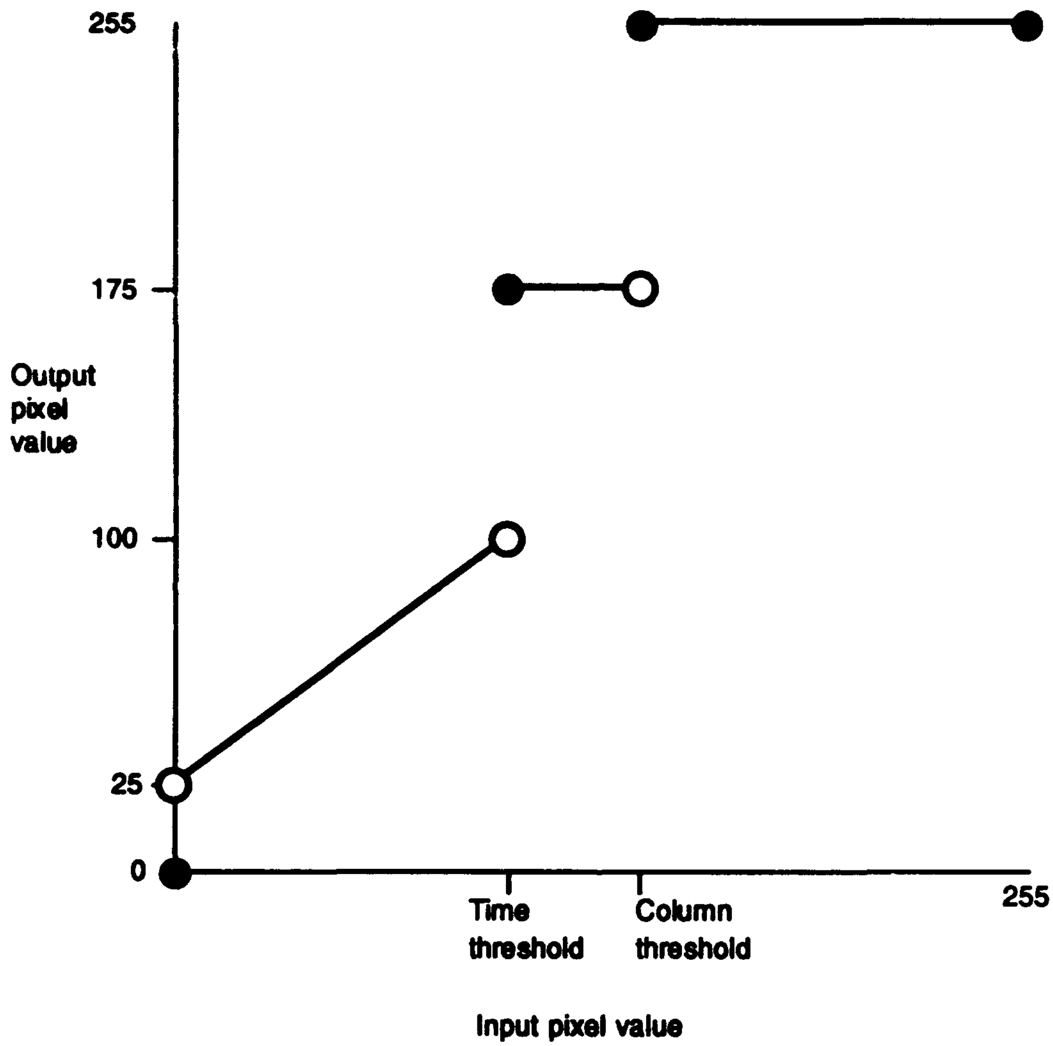

Figure 11 Output LUT for monochrome monitor

Functional

```
col2[LINE_5, CHAR_3] = digit_or_blank(side_col_char(side, COL_2, LINE_5, CHAR_3),
                                      top_col_char(top, LINE_5),
                                      WIDTH_9,
                                      col_cutoff
                                     );
```

Procedural

```
width = 9;
.
.
.
col  = 2;
.
.
.
line = 5;
.
.
.
char = 3;
x = side_col_char(side, col, line, char);
y = top_col_char(top, line);
col2[5, 3] = digit_or_blank (x, y);
```

Figure 12 Optional programming styles

THIS IS A BLANK PAGE

DISTRIBUTION

	No. of Copies
Defence Science and Technology Organisation	
Chief Defence Scientist)	
Central Office Executive)	1
Counsellor, Defence Science, London	Cnt Sht
Counsellor, Defence Science, Washington	Cnt Sht
Senior Defence Scientific Adviser	1
Scientific Adviser, Policy and Command	Cnt Sht
Scientific Adviser, Army	1
Navy Scientific Adviser	1
Air Force Scientific Adviser	1
Director, Trials	1
Electronics Research Laboratory	
Director	1
Chief Information Technology Division	1
Research Leader Command, Control and Intelligence Systems	1
Head Systems Simulation and Assessment	1
Attention: John Coleby	1
Attention: Stephen Bourn	2
Head Exercise Analysis	1
Attention: Paul Rogers	1
Chief Guided Weapons Division	1
Publications and Publicity Officer, Information Technology Division	1
Aeronautical Research Laboratory	
Chief Air Operations Division	1
Defence Force	
Land Headquarters SO Science	1
CO 16 Air Defence Regiment	1
Air Commander Australia	1
Commander Strike Reconnaissance Group	1
OC 82 Wing	1
Maritime Commander Australia	1
Defence Intelligence Organisation	
Assistant Secretary, Scientific Analysis	Cnt Sht
Libraries and Information Services	
Director of Publications, Australian Government Publishing Service	1
OIC, Technical Reports Centre, Defence Central Library Campbell Park	1
Manager, Document Exchange Centre, Defence Information Services	1
National Technical Information Service, United States	2

Defence Research Information Centre, United Kingdom	2
Director Scientific Information Services, Canada	1
Ministry of Defence, New Zealand	1
National Library of Australia	1
Defence Science and Technology Organisation Salisbury, Research Library	2
Library Defence Signals Directorate	1
British Library Document Supply Centre	1
Spares	
Defence Science and Technology Organisation Salisbury, Research Library	6

Department of Defence

DOCUMENT CONTROL DATA SHEET

1. Page Classification: Unclassified			2. Privacy Marking/Caveat (of document)
3a. AR Number: AR-006-924	3b. Laboratory Number: ERL-0593-RN	3c. Type of Report: RESEARCH NOTE	4. Task Number: DST 88/211
5. Document Date: NOVEMBER 1993	6. Cost Code: 779226	7. Security Classification: *U Document, U Title, U Abstract	8. No. of Pages: 28
			9. No. of Refs.: 5

10. Title

REAL TIME CHARACTER RECOGNITION FOR F-111C VIDEO TAPE RECORDS

S (Secret) C (Confi) R (Rest) U (Unclass)

* For UNCLASSIFIED docs with a secondary distribution LIMITATION, use (L) in document box.

11. Author(s)

Stephen Bourn

12. Downgrading/Delimiting Instructions

13a. Corporate Author and Address

Electronics Research Laboratory
PO Box 1500, Salisbury SA 5108

13b. Task Sponsor

DST

14. Officer/Position responsible for

Security:...........N/A...

Downgrading:....N/A..........................   ....

Approval forRelease:...DERL...

15. Secondary Release Statement of this Document

APPROVED FOR PUBLIC RELEASE

16a. Deliberate Announcement

No Limitation

16b. Casual Announcement (for citation in other documents)

[X] No Limitation   [ ] Ref. by Author , Doc No. and date only.

17. DEFTEST Descriptors

Character recognition
F-111C aircraft
Display devices
Military exercises
Videotape recording
Image Processing
Australian Defence Force
Aircraft tracking

18. DISCAT Subject Codes

01

19. Abstract

For evaluation purposes it is often necessary to reconstruct the tracks of F-111C aircraft participating in Australian Defence Force Exercises. The necessary information can be obtained from a video tape recording of one of the aircraft's displays. An image processing system has been developed to automatically extract this data.